

# Curso de L<sup>A</sup>T<sub>E</sub>X

2 de marzo de 2025

## 1 Introducción

## 2 Estructura Básica

- Documento Mínimo
- Paquetes
- Entornos
- Dando Formato al texto

## 3 Listas

## 4 Secciones

# ¿Qué es L<sup>A</sup>T<sub>E</sub>X?

- L<sup>A</sup>T<sub>E</sub>X es un sistema de preparación de documentos.

# ¿Qué es L<sup>A</sup>T<sub>E</sub>X?

- L<sup>A</sup>T<sub>E</sub>X es un sistema de preparación de documentos.
- Utilizado para la creación de documentos científicos y técnicos.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:
  - Un sistema TeX.



# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:
  - Un sistema TeX.
  - Un editor de texto.

- El núcleo del trabajo con L<sup>A</sup>T<sub>E</sub>X es tener disponible un sistema TeX.

- El núcleo del trabajo con  $\text{\LaTeX}$  es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que  $\text{\LaTeX}$  funcione.

- El núcleo del trabajo con L<sup>A</sup>T<sub>E</sub>X es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que L<sup>A</sup>T<sub>E</sub>X funcione.
- Existen dos sistemas TeX principales: MiKTeX y TeX Live. Ambos disponibles para Windows, macOS y Linux.

- El núcleo del trabajo con L<sup>A</sup>T<sub>E</sub>X es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que L<sup>A</sup>T<sub>E</sub>X funcione.
- Existen dos sistemas TeX principales: MiKTeX y TeX Live. Ambos disponibles para Windows, macOS y Linux.
- MiKTeX tiene un fuerte respaldo en Windows; en macOS, TeX Live está incluido en una colección más grande llamada MacTeX.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.



# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.
  - Winedt, un editor comercial para Windows.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.
  - Winedt, un editor comercial para Windows.
  - Overleaf, un editor en línea.



Figura: Distintos editores de L<sub>A</sub>T<sub>E</sub>X.



# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.
- El *argumento* en llaves `{ }` le dice a L<sup>A</sup>T<sub>E</sub>X qué tipo de documento estamos creando: en este ejemplo, `article`.

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}
\begin{document}
    Hello World!
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.
- El *argumento* en llaves `{ }` le dice a L<sup>A</sup>T<sub>E</sub>X qué tipo de documento estamos creando: en este ejemplo, `article`.
- Un signo de porcentaje `%` comienza un *comentario* — L<sup>A</sup>T<sub>E</sub>X ignorará el resto de la línea.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.



# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.
  - `letter`: Cartas.

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.
  - `letter`: Cartas.
  - `beamer`: Presentaciones.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.
  - `twoside`, `oneside`: Impresión a doble o una cara.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.
  - `twoside`, `oneside`: Impresión a doble o una cara.
- Por ejemplo, `\documentclass[12pt,a4paper]{article}` indica que el documento será un artículo con fuente de 12 puntos y tamaño de papel A4.

- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.



- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.
- Se cargan en el preámbulo del documento después del comando `\documentclass`.

- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.
- Se cargan en el preámbulo del documento después del comando `\documentclass`.
- El comando `\usepackage[]{}{}` permite cargar un complemento (plugin), que añade nuevas funcionalidades.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

Ejemplos:

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes T1.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete `inputenc` con la opción `utf8` (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes `T1`.
- `\usepackage[spanish]{babel}`: Carga el paquete `babel`, que se encarga de la tipografía con el idioma español.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes T1.
- `\usepackage[spanish]{babel}`: Carga el paquete babel, que se encarga de la tipografía con el idioma español.
- `\usepackage{graphicx}`: Carga el paquete que permite incluir imágenes externas en el documento.

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.



# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{}` y termina con `\end{}`. Dentro de las `{ }` se especifica el nombre del entorno.

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{}` y termina con `\end{}`. Dentro de las `{ }` se especifica el nombre del entorno.
- El entorno `document` es obligatorio: lo que está dentro constituye el contenido del documento. Fuera del bloque `document`, se encuentran comandos que modifican las características del documento o cómo se imprime (por ejemplo, paquetes o comandos globales).

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{ }` y termina con `\end{ }`. Dentro de las `{ }` se especifica el nombre del entorno.
- El entorno `document` es obligatorio: lo que está dentro constituye el contenido del documento. Fuera del bloque `document`, se encuentran comandos que modifican las características del documento o cómo se imprime (por ejemplo, paquetes o comandos globales).
- Todos los demás entornos son opcionales y se usan según sea necesario.

**Ejemplo:** El entorno `itemize` crea listas con viñetas (listas sin numerar). Por lo tanto, una lista con viñetas se crea cada vez que se llama al comando `\item`.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**



# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**
- Formato de tamaño de fuente: `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**
- Formato de tamaño de fuente: `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`.
- Forzar un salto de línea: `\\`.

# Dando Formato al texto

El entorno `itemize` crea listas con viñetas (listas sin numerar).

```
\begin{itemize}  
  \item Elemento 1  
  \item Elemento 2  
  \item Elemento 3  
\end{itemize}
```

- Elemento 1
- Elemento 2
- Elemento 3

# Dando Formato al texto

El entorno `enumerate` crea listas numeradas.

```
\begin{enumerate}  
\item Elemento 1  
\item Elemento 2  
\item Elemento 3  
\end{enumerate}
```

- 1 Elemento 1
- 2 Elemento 2
- 3 Elemento 3

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.



LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.
- `\subparagraph{texto}`: Subpárrafo.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.
- `\subparagraph{texto}`: Subpárrafo.
- `\chapter{texto}`: Capítulo.

```
\documentclass{article}  
\usepackage[T1]{fontenc}  
\begin{document}  
  Hola Mundo!
```

Primer Documento.

```
\section{Primera Sección}  
Texto de la primera sección.
```

Segundo Párrafo.

```
\subsection{Subseccion de la primera sección}  
Texto de la subsección.
```

```
\subsubsection{Subsubsección de la primera sección}  
Texto de la subsubsección.
```

```
\end{document}
```

Hola Mundo!  
Primer Documento.

## 1 Primera Seccion

Texto de la seccion.  
Segundo parrafo.

### 1.1 Subseccion de la primera seccion

Texto de la subseccion.

#### 1.1.1 Subsubseccion de la primera seccion

Texto de la subsubseccion.

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ``` sobre el lado izquierdo y el apóstrofe `'` sobre el lado derecho.  
Comillas simple: `'texto'`.  
Comillas dobles: `"texto"`.

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ``` sobre el lado izquierdo y el apóstrofe `'` sobre el lado derecho.

Comillas simple: `'texto'`.

Comillas dobles: `"texto"`.

- Algunos caracteres comunes tienen significados especiales en  $\text{\LaTeX}$ :

`%` Signo de porcentaje

`#` Signo numeral

`&` Ampersand

`$` Signo pesos

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ` sobre el lado izquierdo y el apóstrofe ' sobre el lado derecho.

Comillas simple: ‘texto’.

Comillas dobles: “texto”.

- Algunos caracteres comunes tienen significados especiales en L<sup>A</sup>T<sub>E</sub>X:

% Signo de porcentaje

# Signo numeral

& Ampersand

\$ Signo pesos

- Si son usados, tendremos errores en la compilación. Para usar estos caracteres en la salida, se debe colocar barra invertida al caracter.

`\$ \% \& \#!`

# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.



# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.

# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.

# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.

# Tipografía Matemática

- ¿Por qué son especiales los signos  $\$$ ? Los usamos para marcar contenido matemático en el texto.

% no tan bueno:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

% mucho mejor:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

# Tipografía Matemática

- ¿Por qué son especiales los signos  $\$$ ? Los usamos para marcar contenido matemático en el texto.

% no tan bueno:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

% mucho mejor:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

- Utilice siempre los signos de pesos en pares — uno para comenzar el contenido matemático, y uno para terminarlo.

# Tipografía Matemática

- ¿Por qué son especiales los signos pesos \$? Los usamos para marcar contenido matemático en el texto.

<p>% no tan bueno:</p> <p>Sean <math>a</math> y <math>b</math> tales que <math>c = a - b + 1</math>.</p> <p>% mucho mejor:</p> <p>Sean \$<math>a</math>\$ y \$<math>b</math>\$ tales que \$<math>c = a - b + 1</math>\$.</p>	<p>Sean <math>a</math> y <math>b</math> tales que <math>c = a - b + 1</math>.</p> <p>Sean <math>a</math> y <math>b</math> tales que <math>c = a - b + 1</math>.</p>
--	---

- Utilice siempre los signos de pesos en pares — uno para comenzar el contenido matemático, y uno para terminarlo.
- L<sup>A</sup>T<sub>E</sub>X maneja el espacio automáticamente; por lo que ignorará los que se hayan puesto.

Sea \$ $y=mx+b$ \$ ...	Sea $y = mx + b$ ...
Sea \$ $y = m x + b$ \$ ...	Sea $y = mx + b$ ...

# Tipografía Matemática:Notación

- Use el signo  $\wedge$  para indicar superíndices y el guión bajo  $_$  para marcar subíndices.

$y = c_2 x^2 + c_1 x + c_0$	$y = c_2 x^2 + c_1 x + c_0$
-----------------------------	-----------------------------

# Tipografía Matemática:Notación

- Use el signo  $\wedge$  para indicar superíndices y el guión bajo  $_$  para marcar subíndices.

$y = c_2 x^2 + c_1 x + c_0$	$y = c_2 x^2 + c_1 x + c_0$
-----------------------------	-----------------------------

- Utilice las llaves  $\{ \}$  para agrupar superíndices y subíndices.

$F_n = F_{n-1} + F_{n-2}$ % oops!	$F_n = F_n - 1 + F_n - 2$
$F_n = F_{\{n-1\}} + F_{\{n-2\}}$ % ok!	$F_n = F_{n-1} + F_{n-2}$



# Tipografía Matemática:Notación

- Use el signo  $\wedge$  para indicar superíndices y el guión bajo  $_$  para marcar subíndices.

<code>\$y = c_2 x^2 + c_1 x + c_0\$</code>	$y = c_2 x^2 + c_1 x + c_0$
--	-----------------------------

- Utilice las llaves  $\{ \}$  para agrupar superíndices y subíndices.

<code>\$F_n = F_{n-1} + F_{n-2}\$% oops!</code>	$F_n = F_n - 1 + F_n - 2$
<code>\$F_n = F_{\{n-1\}} + F_{\{n-2\}}\$% ok!</code>	$F_n = F_{n-1} + F_{n-2}$

- Hay comandos para letras Griegas y notación común.

<code>\$\mu = A e^{\{Q/RT\}}\$</code>	$\mu = A e^{Q/RT}$
<code>\$\Omega = \sum_{\{k=1\}}^n \omega_k\$</code>	$\Omega = \sum_{k=1}^n \omega_k$

# Tipografía Matemática: Entornos

- `equation` es un *entorno*.

# Tipografía Matemática: Entornos

- `equation` es un *entorno*.
- Un comando puede producir diferentes salidas en diferentes contextos.

Podemos escribir `\Omega = \sum_{k=1}^n \omega_k` en nuestro texto, o podemos escribir `\begin{equation}`  
`\Omega = \sum_{k=1}^n \omega_k`  
`\end{equation}` para mostrarlo en un entorno diferente.

Podemos escribir  $\Omega = \sum_{k=1}^n \omega_k$  en nuestro texto, o podemos escribir

$$\Omega = \sum_{k=1}^n \omega_k \quad (1)$$

para mostrarlo en un entorno diferente.

# Tipografía Matemática: Entornos

- `equation` es un *entorno*.
- Un comando puede producir diferentes salidas en diferentes contextos.

Podemos escribir `\Omega = \sum_{k=1}^n \omega_k` en nuestro texto, o podemos escribir `\begin{equation}`  
`\Omega = \sum_{k=1}^n \omega_k` `\end{equation}`  
para mostrarlo en un entorno diferente.

Podemos escribir  $\Omega = \sum_{k=1}^n \omega_k$  en nuestro texto, o podemos escribir

$$\Omega = \sum_{k=1}^n \omega_k \quad (1)$$

para mostrarlo en un entorno diferente.

- Note como el  $\Sigma$  es más grande en el entorno `equation`, y como el subíndice y superíndice cambian de posición, a pesar de que utilizamos los mismos comandos.

# Ejemplos con `amsmath`

- Utilice `equation*` (“ecuación-asterisco”) para ecuaciones no-numeradas.

```
\begin{equation*}
```

```
\Omega = \sum_{k=1}^n
```

```
\omega_k
```

```
\end{equation*}
```

$$\Omega = \sum_{k=1}^n \omega_k$$

# Ejemplos con `amsmath`

- Utilice `equation*` (“ecuación-asterisco”) para ecuaciones no-numeradas.

```
\begin{equation*}
```

```
\Omega = \sum_{k=1}^n
```

```
\omega_k
```

```
\end{equation*}
```

$$\Omega = \sum_{k=1}^n \omega_k$$

- $\text{\LaTeX}$  trata las letras adyacentes como variables multiplicadas entre sí, lo cual no siempre es lo que se quiere. `amsmath` define comandos para muchos operadores matemáticos comunes.

```
\begin{equation*} \% \text{ bad!}
```

```
\min_{x,y} (1-x)^2
```

```
\end{equation*}
```

```
\begin{equation*} \% \text{ good!}
```

```
\min_{x,y} \{(1-x)^2\}
```

```
\end{equation*}
```

$$\min_{x,y} (1-x)^2$$

$$\min_{x,y} (1-x)^2$$

# Ejemplos con `amsmath`

- Alinear una secuencia de ecuaciones al signo igual

$$\begin{aligned}(x+1)^3 &= (x+1)(x+1)(x+1) \\ &= (x+1)(x^2+2x+1) \\ &= x^3+3x^2+3x+1\end{aligned}$$

con el entorno `align*`.

# Ejemplos con `amsmath`

- Alinear una secuencia de ecuaciones al signo igual

$$\begin{aligned}(x+1)^3 &= (x+1)(x+1)(x+1) \\ &= (x+1)(x^2+2x+1) \\ &= x^3+3x^2+3x+1\end{aligned}$$

con el entorno `align*`.

- El ampersand `&` separa la columna izquierda (antes del `=`) de la columna derecha (después del `=`).

```
\begin{align*}
(x+1)^3 &= (x+1)(x+1)(x+1) \\
&= (x+1)(x^2+2x+1) \\
&= x^3+3x^2+3x+1
\end{align*}
```



# Ejemplos con `amsmath`

- Alinear una secuencia de ecuaciones al signo igual

$$\begin{aligned}(x+1)^3 &= (x+1)(x+1)(x+1) \\ &= (x+1)(x^2+2x+1) \\ &= x^3+3x^2+3x+1\end{aligned}$$

con el entorno `align*`.

- El ampersand `&` separa la columna izquierda (antes del `=`) de la columna derecha (después del `=`).

```
\begin{align*}
(x+1)^3 &= (x+1)(x+1)(x+1) \\
&= (x+1)(x^2+2x+1) \\
&= x^3+3x^2+3x+1
\end{align*}
```

- Una doble barra invertida `\\` da comienzo a una nueva línea.

# Etiquetas y Referencias Cruzadas

- Utilice `label` y `ref` para la numeración automática.

# Etiquetas y Referencias Cruzadas

- Utilice `label` y `ref` para la numeración automática.
- El paquete `amsmath` proporciona `eqref` para las referencias de ecuaciones.

# Etiquetas y Referencias Cruzadas

- Utilice `label` y `ref` para la numeración automática.
- El paquete `amsmath` proporciona `eqref` para las referencias de ecuaciones.

```
\documentclass{article}
\usepackage{amsmath} %      para
\eqref
```

```
\begin{document}
```

```
\section{Introducción}
\label{sec:intro}
```

## 1 Introducción

En la Sección 2, we ...

```
En la Sección \ref{sec:metodo},
hemos ...
```

## 2 Método

```
\section{Método}
\label{sec:metodo}
```

$$e^{i\pi} + 1 = 0 \tag{1}$$

```
\begin{equation}
\label{eq:euler}
e^{i\pi} + 1 = 0
\end{equation}
```

Por (1), Tenemos ...

```
Por \eqref{eq:euler}, Tenemos ...
```

```
\end{document}
```

# Usando Imágenes

- Para importar imágenes en  $\text{\LaTeX}$ , se usa el paquete `graphicx`, que añade el comando `includegraphics` a  $\text{\LaTeX}$ .

# Usando Imágenes

- Para importar imágenes en L<sup>A</sup>T<sub>E</sub>X, se usa el paquete `graphicx`, que añade el comando `includegraphics` a L<sup>A</sup>T<sub>E</sub>X.
- El comando `includegraphics` toma un argumento obligatorio, que es el nombre del archivo de la imagen.

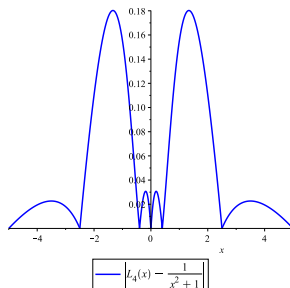
# Usando Imágenes

- Para importar imágenes en  $\text{\LaTeX}$ , se usa el paquete `graphicx`, que añade el comando `includegraphics` a  $\text{\LaTeX}$ .
- El comando `includegraphics` toma un argumento obligatorio, que es el nombre del archivo de la imagen.
- Se puede insertar ficheros EPS, PNG, JPG y PDF. Si dispone de varias versiones de la imagen entonces se puede escribir la extensión.

# Usando Imágenes

- Para importar imágenes en  $\text{\LaTeX}$ , se usa el paquete `graphicx`, que añade el comando `includegraphics` a  $\text{\LaTeX}$ .
- El comando `includegraphics` toma un argumento obligatorio, que es el nombre del archivo de la imagen.
- Se puede insertar ficheros EPS, PNG, JPG y PDF. Si dispone de varias versiones de la imagen entonces se puede escribir la extensión.

```
\usepackage{graphicx}
\begin{document}
\includegraphics{imagen.png}
\end{document}
```





# Cambiando la apariencia de la imagen

- El comando `includegraphics` tiene opciones para cambiar la apariencia de la imagen.

# Cambiando la apariencia de la imagen

- El comando `includegraphics` tiene opciones para cambiar la apariencia de la imagen.
- Opciones para controlar el tamaño y la forma de las imágenes incluidas, pudiendo incluso recortarlas.

# Cambiando la apariencia de la imagen

- El comando `includegraphics` tiene opciones para cambiar la apariencia de la imagen.
- Opciones para controlar el tamaño y la forma de las imágenes incluidas, pudiendo incluso recortarlas.
- La opción más sencilla es definir el anchura y alto de una imagen, las cuales se pueden dar de forma relativa con respecto al ancho `\textwidth` y al alto `\textheight` de la zona de texto.

# Cambiando la apariencia de la imagen

- El comando `includegraphics` tiene opciones para cambiar la apariencia de la imagen.
- Opciones para controlar el tamaño y la forma de las imágenes incluidas, pudiendo incluso recortarlas.
- La opción más sencilla es definir el anchura y alto de una imagen, las cuales se pueden dar de forma relativa con respecto al ancho `\textwidth` y al alto `\textheight` de la zona de texto.
- $\text{\LaTeX}$  ajustará la escala de la imagen automáticamente para que la proporción de las dimensiones de la imagen sea la correcta.

# Cambiando la apariencia de la imagen

- El comando `includegraphics` tiene opciones para cambiar la apariencia de la imagen.
- Opciones para controlar el tamaño y la forma de las imágenes incluidas, pudiendo incluso recortarlas.
- La opción más sencilla es definir el anchura y alto de una imagen, las cuales se pueden dar de forma relativa con respecto al ancho `\textwidth` y al alto `\textheight` de la zona de texto.
- $\text{\LaTeX}$  ajustará la escala de la imagen automáticamente para que la proporción de las dimensiones de la imagen sea la correcta.
- Para cambiar el ancho se puede usar `\includegraphics[width=0.5\textwidth]{imagen.png}`.

# Cambiando la apariencia de la imagen

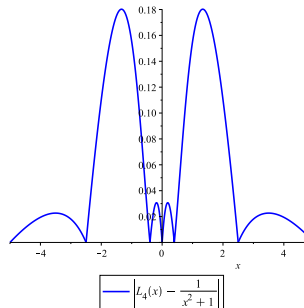
- El comando `\includegraphics` tiene opciones para cambiar la apariencia de la imagen.
- Opciones para controlar el tamaño y la forma de las imágenes incluidas, pudiendo incluso recortarlas.
- La opción más sencilla es definir el anchura y alto de una imagen, las cuales se pueden dar de forma relativa con respecto al ancho `\textwidth` y al alto `\textheight` de la zona de texto.
- $\text{\LaTeX}$  ajustará la escala de la imagen automáticamente para que la proporción de las dimensiones de la imagen sea la correcta.
- Para cambiar el ancho se puede usar `\includegraphics[width=0.5\textwidth]{imagen.png}`.
- Para cambiar la altura, se puede usar `\includegraphics[height=0.5\textheight]{imagen.png}`.

# Cambiando la apariencia de la imagen

```
\usepackage{graphicx}
\begin{document}
\includegraphics[width=0.25\textwidth]{imagen.png}
\end{document}
```



```
\usepackage{graphicx}
\begin{document}
\includegraphics[height=0.75\textheight]{imagen.png}
\end{document}
```



# Cambiando la apariencia de la imagen

- También se puede cambiar la escala `scale` de las imágenes o hacerlas rotar de un ángulo dado con `angle`.



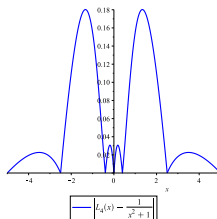
# Cambiando la apariencia de la imagen

- También se puede cambiar la escala `scale` de las imágenes o hacerlas rotar de un ángulo dado con `angle`.
- La otra cosa que puede hacer es recortar una imagen con `clip` y `trim`.

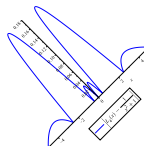
# Cambiando la apariencia de la imagen

- También se puede cambiar la escala **scale** de las imágenes o hacerlas rotar de un ángulo dado con **angle**.
- La otra cosa que puede hacer es recortar una imagen con **clip** y **trim**.

```
\usepackage{graphicx}
\begin{document}
\includegraphics[scale=0.15]{imagen.png}
\end{document}
```



```
\usepackage{graphicx}
\begin{document}
\includegraphics[angle=45]{imagen.png}
\end{document}
```



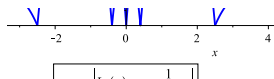
# Cambiando la apariencia de la imagen

- La otra cosa que puede hacer es recortar una imagen con `clip` y `trim`.

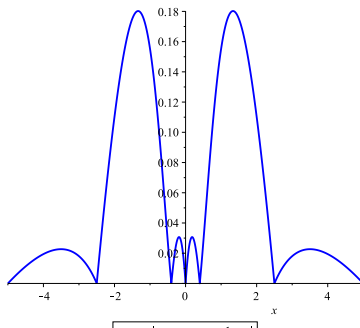
# Cambiando la apariencia de la imagen

- La otra cosa que puede hacer es recortar una imagen con `clip` y `trim`.

```
\usepackage{graphicx}
\begin{document}
\includegraphics[clip,trim=100 50 50
400]{imagen.png}
\end{document}
```



```
\usepackage{graphicx}
\begin{document}
\includegraphics[trim=0cm 0cm 0cm
0cm]{imagen.png}
\end{document}
```



# Insertando Tablas

- El paquete `array` añade más funcionalidades a las dadas por  $\text{\LaTeX}$  y que no forma parte del núcleo de  $\text{\LaTeX}$ .

# Insertando Tablas

- El paquete `array` añade más funcionalidades a las dadas por  $\text{\LaTeX}$  y que no forma parte del núcleo de  $\text{\LaTeX}$ .
- Para insertar tablas en  $\text{\LaTeX}$ , se usa el comando `tabular`.

# Insertando Tablas

- El paquete `array` añade más funcionalidades a las dadas por  $\text{\LaTeX}$  y que no forma parte del núcleo de  $\text{\LaTeX}$ .
- Para insertar tablas en  $\text{\LaTeX}$ , se usa el comando `tabular`.
- El comando `tabular` toma un argumento obligatorio, que es el alineamiento de las columnas.

# Insertando Tablas

- El paquete `array` añade más funcionalidades a las dadas por  $\text{\LaTeX}$  y que no forma parte del núcleo de  $\text{\LaTeX}$ .
- Para insertar tablas en  $\text{\LaTeX}$ , se usa el comando `tabular`.
- El comando `tabular` toma un argumento obligatorio, que es el alineamiento de las columnas.

Tipo	Descripción
<code>l</code>	columna justificada a la izquierda
<code>c</code>	columna centrada
<code>r</code>	columna justificada a la derecha
<code>p{width}</code>	columna con un ancho <code>width</code> fijo; el texto sera automáticamente ajustado a la línea y justificado
<code>m{width}</code>	como <code>p</code> , pero centrado verticalmente con respecto al resto del texto de la misma fila
<code>b{width}</code>	como <code>p</code> , pero ajustado verticalmente a la parte baja de la celda
<code>w{align}{width}</code>	imprime el contenido con un ancho <code>width</code> fijo, sobreimprimiendo si el texto es muy largo. Puede elegir el justificado horizontal <code>align</code> usando <code>l</code> , <code>c</code> , or <code>r</code> .
<code>W{align}{width}</code>	como <code>w</code> , pero dando lugar a un mensaje de alerta «overfull box warning» si el texto es demasiado grande.



# Insertando Tablas

- Las columnas `c` y `l` y `r` tendrán la anchura de la celda más ancha.

# Insertando Tablas

- Las columnas `c` y `l` y `r` tendrán la anchura de la celda más ancha.
- Cada columna debe ser declarada, con lo que si se quiere tres columnas centradas, se tendrá que usar `ccc` en el preámbulo de la tabla. Los espacios son ignorados, con lo que `c c c` tendrá el mismo efecto.

# Insertando Tablas

- Las columnas `c` y `l` y `r` tendrán la anchura de la celda más ancha.
- Cada columna debe ser declarada, con lo que si se quiere tres columnas centradas, se tendrá que usar `ccc` en el preámbulo de la tabla. Los espacios son ignorados, con lo que `c c c` tendrá el mismo efecto.
- En el cuerpo de la tabla, las columnas se separan usando el símbolo `&` y una nueva fila comienza con los símbolos `\\`.

# Insertando Tablas

```
\usepackage{array}
\begin{document}
\begin{tabular}{lll}
Animal & Comida & Tamaño \\
perro & carne & mediano \\
caballo & heno & grande \\
rana & moscas & pequeño
\end{tabular}
\end{document}
```

Animal	Comida	Tamaño
perro	carne	mediano
caballo	heno	grande
rana	moscas	pequeño

# Insertando Tablas

- Si la columna de una tabla contine mucho texto, se tendrá problemas al colocarlo únicamente con `l`, `r` o `c`.

# Insertando Tablas

- Si la columna de una tabla contine mucho texto, se tendrá problemas al colocarlo únicamente con `l`, `r` o `c`.

```
\usepackage{array}
\begin{document}
\begin{tabular}{cl}
Animal & Descripción \\
perro & El perro es un miembro del
género Canis, el cual forma parte de los
cánidos derivados del lobo y es uno de
los carnívoros terrestres más
abundantes. \\
gato & El gato es una especie doméstica
de pequeños mamíferos carnívoros. Es la
única especie domesticada de la familia
de los félidos y es a menudo llamado
gato doméstico, para diferenciarlo de
los miembros salvajes de la familia. \\
\end{tabular}
\end{document}
```

Animal	Descripción
perro	El perro es un miembro del género Canis, el cual forma parte de los cánidos derivados del lobo y es uno de los carnívoros terrestres más abundantes.
gato	El gato es una especie doméstica de pequeños mamíferos carnívoros. Es la única especie domesticada de la familia de los félidos y es a menudo llamado gato doméstico, para diferenciarlo de los miembros salvajes de la familia.

# Insertando Tablas

- El problema, es que el tipo de columna escribe el contenido de la celda en una única fila con el ancho que le corresponde, aunque haya un borde de página de por medio.

# Insertando Tablas

- El problema, es que el tipo de columna escribe el contenido de la celda en una única fila con el ancho que le corresponde, aunque haya un borde de página de por medio.
- Para evitar este problema se puede usar la columna `p{ancho}` especificando el ancho como argumento.



# Insertando Tablas

- El problema, es que el tipo de columna escribe el contenido de la celda en una única fila con el ancho que le corresponde, aunque haya un borde de página de por medio.
- Para evitar este problema se puede usar la columna `p{ancho}` especificando el ancho como argumento.
- También se puede usar el comando `b{ancho},m{ancho},w{ancho}`.

# Insertando Tablas

```
\usepackage{array}
\begin{document}
\begin{tabular}{cp{4cm}}
Animal & Descripción \\
perro & El perro es un miembro del
género Canis, el cual forma parte de los
cánidos derivados del lobo y es uno de
los carnívoros terrestres más
abundantes. \\
gato & El gato es una especie doméstica
de pequeños mamíferos carnívoros. Es la
única especie domesticada de la familia
de los félidos y es a menudo llamado
gato doméstico, para diferenciarlo de
los miembros salvajes de la familia. \\
\end{tabular}
\end{document}
```

Animal	Descripción
perro	El perro es un miembro del género Canis, el cual forma parte de los cánidos derivados del lobo y es uno de los carnívoros terrestres más abundantes.
gato	El gato es una especie doméstica de pequeños mamíferos carnívoros. Es la única especie domesticada de la familia de los félidos y es a menudo llamado gato doméstico, para diferenciarlo de los miembros salvajes de la familia.

# Insertando Tablas

- El entorno `tabular` proporciona flexibilidad adicional; por ejemplo, se puede insertar líneas divisorias entre cada columnam, para separar las columnas.

# Insertando Tablas

- El entorno `tabular` proporciona flexibilidad adicional; por ejemplo, se puede insertar líneas divisorias entre cada columnam, para separar las columnas.
- Se puede usar el comando `hline` para insertar una línea horizontal.

# Insertando Tablas

- El entorno `tabular` proporciona flexibilidad adicional; por ejemplo, se puede insertar líneas divisorias entre cada columnam, para separar las columnas.
- Se puede usar el comando `hline` para insertar una línea horizontal.
- Se puede usar el comando `cline` para insertar una linea con un estilo diferente.

# Insertando Tablas

- El entorno `tabular` proporciona flexibilidad adicional; por ejemplo, se puede insertar líneas divisorias entre cada columnam, para separar las columnas.
- Se puede usar el comando `hline` para insertar una línea horizontal.
- Se puede usar el comando `cline` para insertar una línea con un estilo diferente.
- El paquete `booktabs` proporciona otras opciones para personalizar las tablas.